

FIG. 1

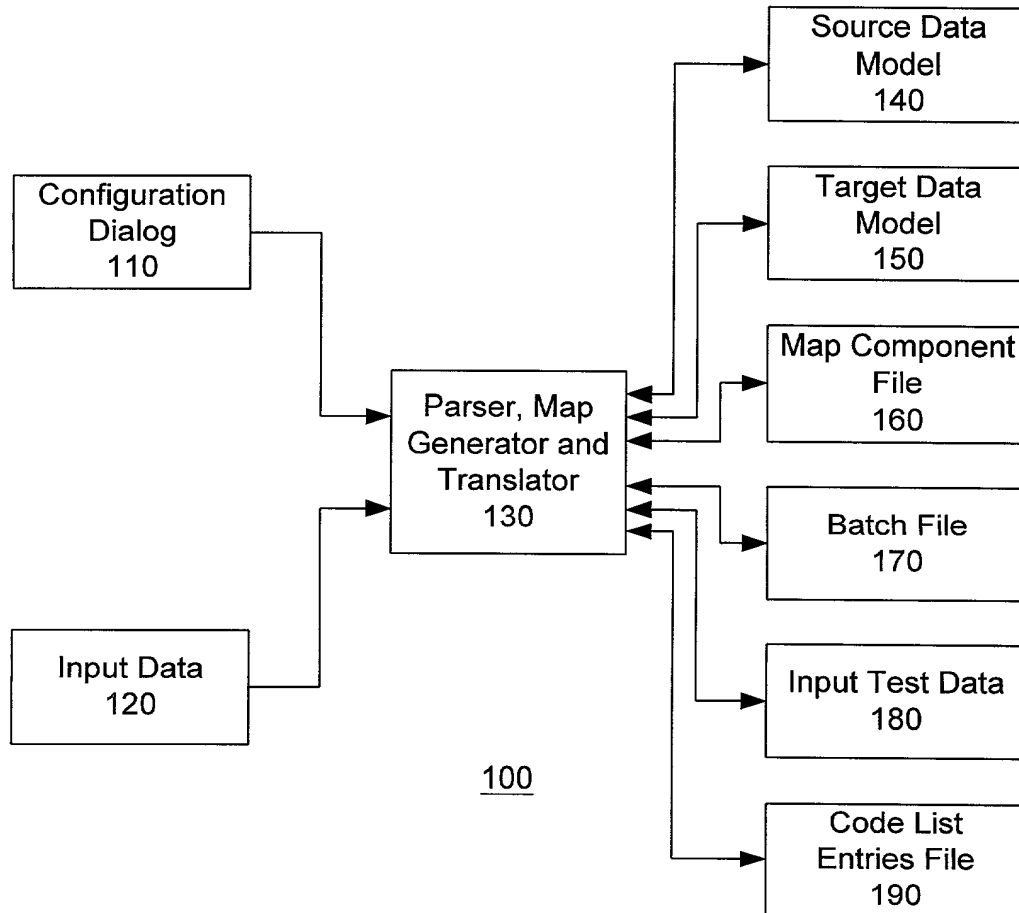


FIG. 2

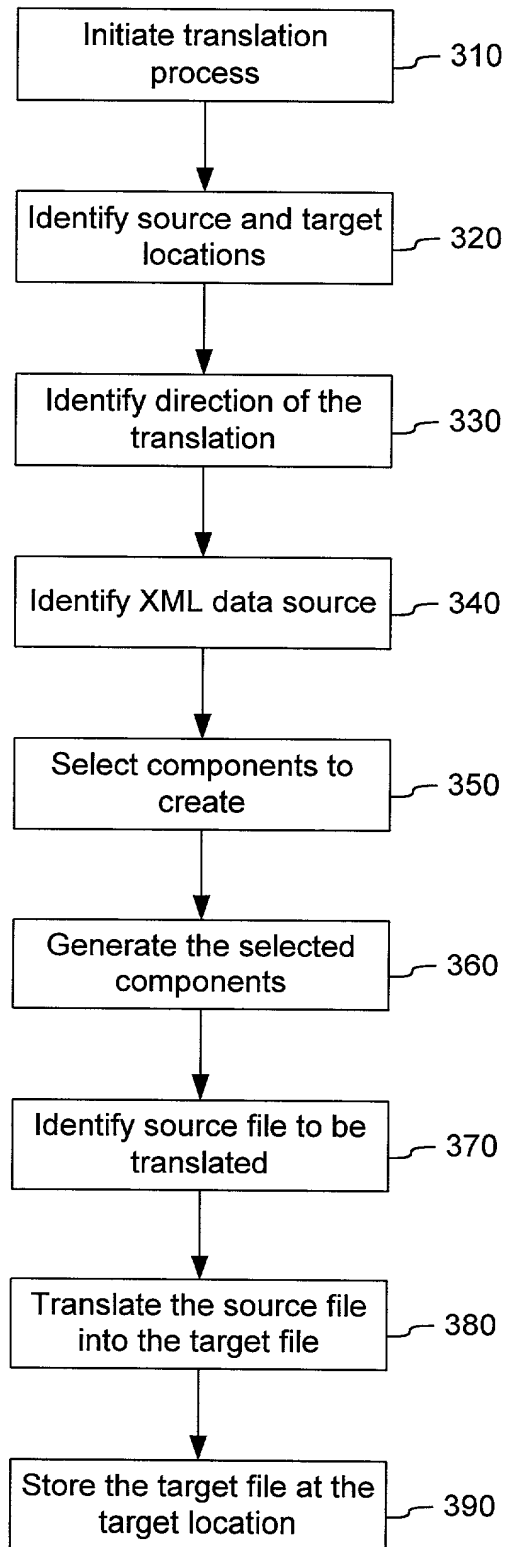


FIG. 3

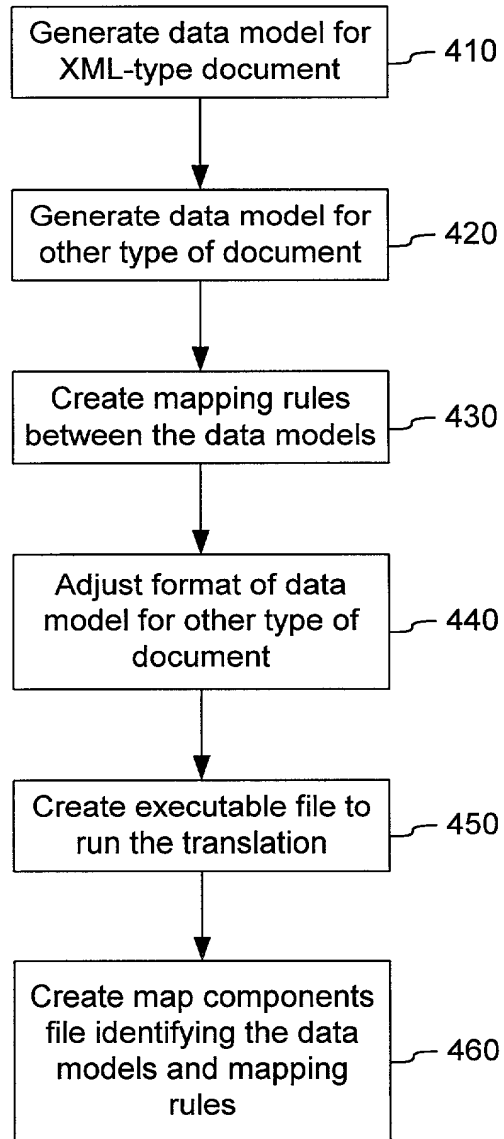


FIG. 4

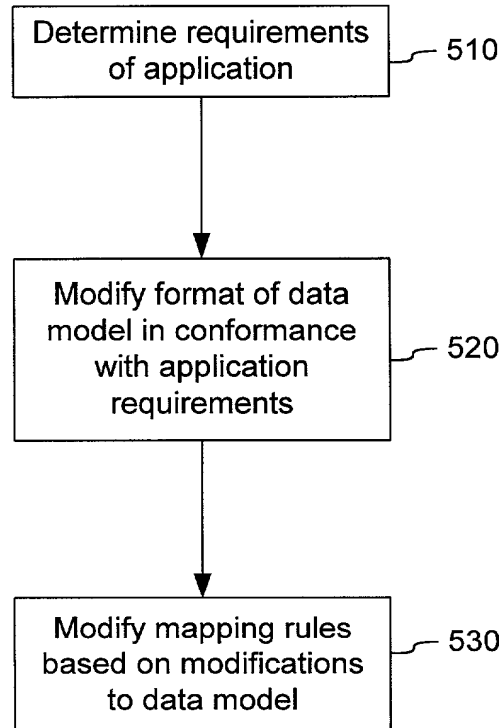


FIG. 5

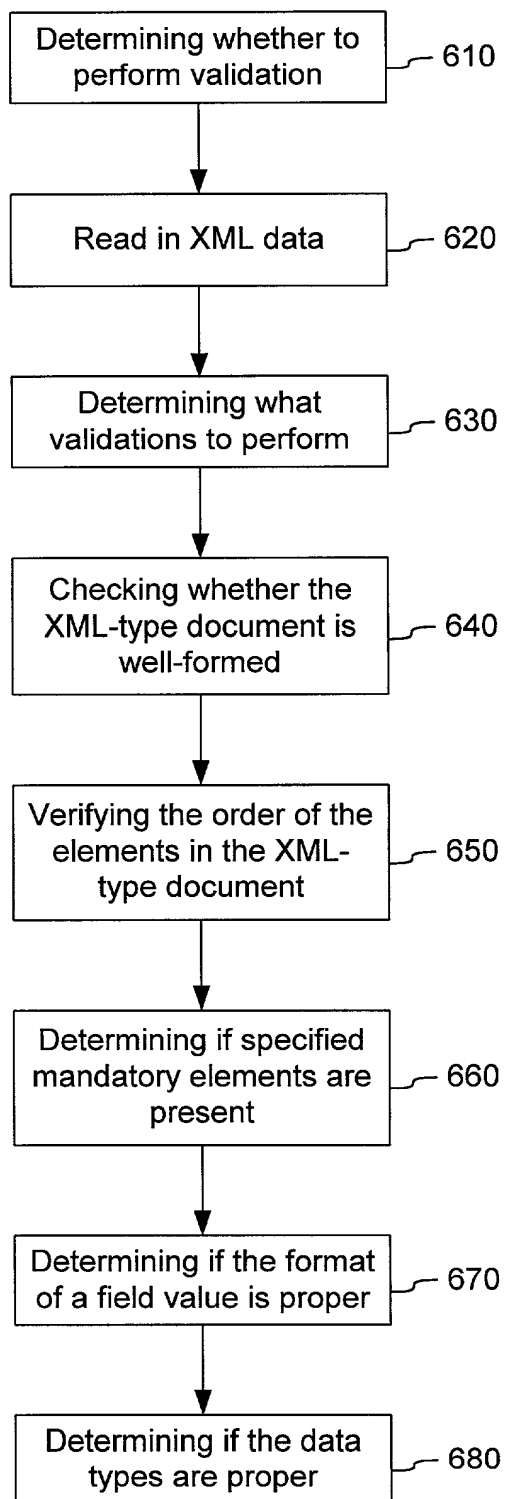


FIG. 6

Title: SYSTEM AND METHOD FOR  
TRANSFORMING DOCUMENTS TO AND  
FROM AN XML FORMAT

Inventor(s): Matthew ROZEK et al.  
Atty. Dkt. No. 088305-0140

```
<xsd:schema xmlns:xsd="http://www.w3.org/2001/XMLSchema" --- 700
  xmlns="test"
  targetNamespace="test"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://www.w3.org/2001/XMLSchema
    http://www.w3.org/2001/XMLSchema.xsd">

  <xsd:element name="purchaseOrder" type="PurchaseOrderType"/> --- 702

  <xsd:complexType name="PurchaseOrderType"> --- 704
    <xsd:sequence> --- 706
      <xsd:element name="PONumber" type="xsd:string"/> --- 708
      <xsd:element name="PODate" type="xsd:date"/> --- 710
      <xsd:element name="ShipToName" type="xsd:string"/> --- 712
      <xsd:element name="ShipToCode" type="xsd:string"/> --- 714
      <xsd:element name="items" type="Items"/> --- 716
    </xsd:sequence> --- 718
  </xsd:complexType> --- 720

  <xsd:complexType name="Items"> --- 722
    <xsd:sequence>
      <xsd:element name="item" minOccurs="0" maxOccurs="unbounded"> --- 724
        <xsd:complexType>
          <xsd:sequence>
            <xsd:element name="productName" type="xsd:string"/>
            <xsd:element name="productNumber" type="xsd:string"/>
            <xsd:element name="quantity">
              <xsd:simpleType>
                <xsd:restriction base="xsd:positiveInteger">
                  <xsd:maxExclusive value="500"/>
                </xsd:restriction>
              </xsd:simpleType>
            </xsd:element>
            <xsd:element name="USPrice" type="xsd:decimal"/>
            <xsd:element name="shipDate" type="xsd:date" minOccurs="0"/>
          </xsd:sequence>
        </xsd:complexType>
      </xsd:element>
    </xsd:sequence>
  </xsd:complexType>

</xsd:schema>
```

FIG. 7

```
;;; (type = SOURCE)
;;; (access = OTXMLS.acc)
DECLARATIONS { --- 800
INCLUDE "OTMainInit.inc"
INCLUDE "OTXML.inc"
INCLUDE "User.inc"
}
Initialization { --- 802
[]
VAR->OTMdlVer = "StdVer 3.2 - Do Not Remove This Line"
PERFORM("OTSrcInit", "STD|XML|(null)~purchaseOrder~(null)", "XML") --- 804
ON_ERROR("xmlOnError")
[]
SET_FIRST_DELIM(62) ; >
SET_SECOND_DELIM(60) ; <
SET_THIRD_DELIM(47) ; /
SET_FOURTH_DELIM(34) ; "
SET_FIFTH_DELIM(39) ; '
SET_RELEASE(92) ; \
}*1 .. 1 ;;|-- end Initialization --|

XMLVersionTag { Prolog "?xml" --- 806
XMLVersionList { PrologList
XMLVersion { Attribute "version" --- 808
XMLVersion_Fld { AttributeVal @1 .. 4092 none --- 810
[]
VAR->OTXMLVersion = XMLVersion_Fld
}*1 .. 1 ;; |-- end XMLVersion_Fld --|
}*0 .. 1 ;; |-- end XMLVersion --| --- 812
Encoding { Attribute "encoding" --- 814
Encoding_Fld { AttributeVal @1 .. 4092 none --- 816
[]
VAR->OTEncoding = Encoding_Fld
}*1 .. 1 ;; |-- end Encoding_Fld --|
}*0 .. 1 ;; |-- end Encoding --|
}*0 .. 1 ;; |-- end XMLVersionList --|
}*0 .. 1 ;; |-- end XMLVersionTag --|
purchaseOrderElement { AttributeElement "purchaseOrder" --- 818
purchaseOrderList { AttributeList --- 820
List_grp {
List_label { AttributeLabel @1 .. 255 none
:ERROR
[] ;; Necessary rule to turn off ON_ERROR()
REJECT
}*1 .. 1 ;; |-- end List_label --|
List_value { AttributeVal @1 .. 4092 none
}*1 .. 1 ;; |-- end List_value --|
}*0 .. 50 ;; |-- end List_grp --| --- 822
}*1 .. 1 ;; |-- end purchaseOrderList --|
SeqGroup {
```

FIG. 8